

Make Decisions with Decision Trees



Input

- x
- Set of attributes
- Describe data or situation

Output

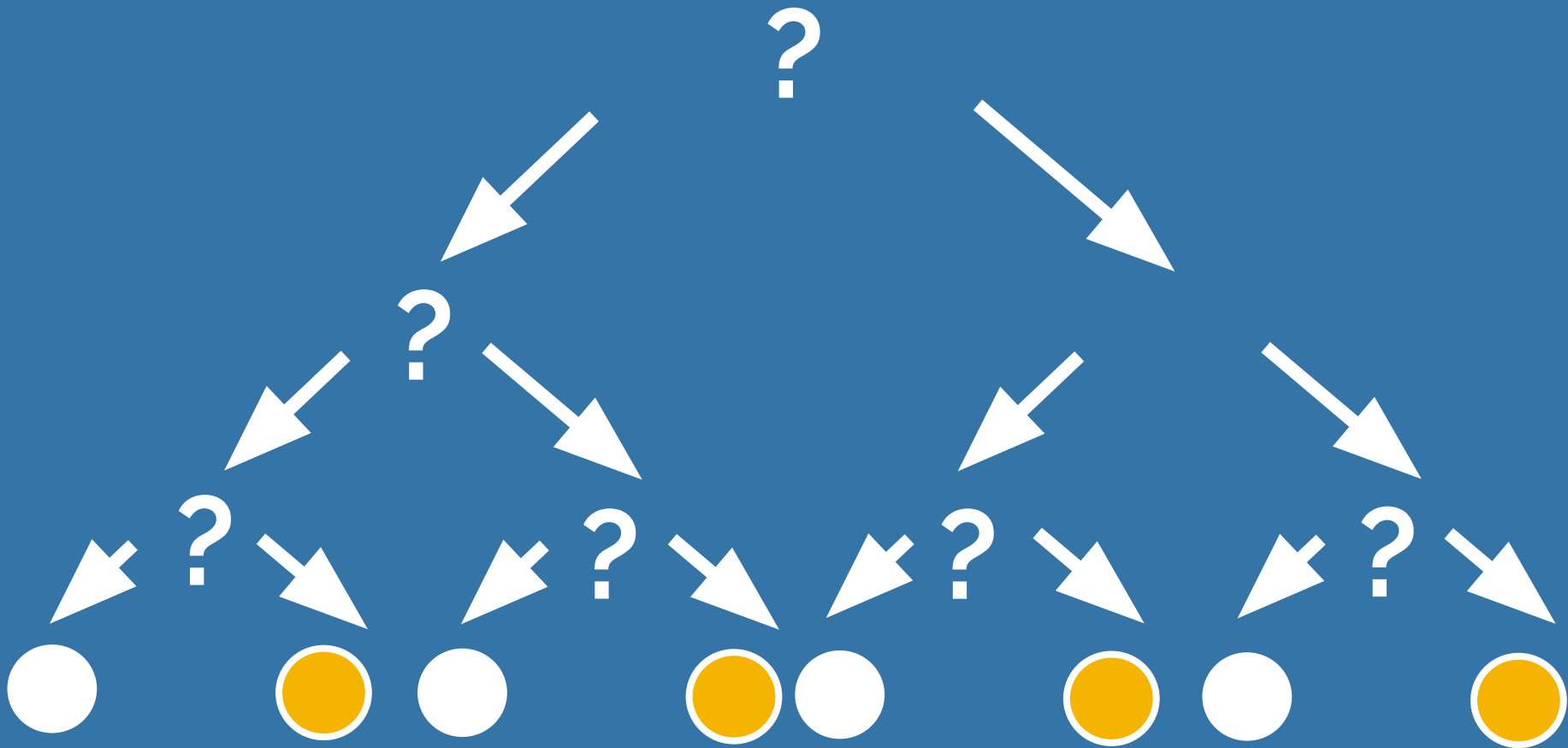
- y
- Decision
- Predicted value for input

Decision Tree

- Input: Vectors of attributes
- Output: Yes/No based on goal

Decision Tree Example

- **Goal: Wait for an available table?**
- **Input:**
 - How busy is the restaurant?
 - How long is the wait?
 - Am I hungry?
 - Did I make a reservation?
 - ...



Problem

- Some problems require decision trees that are too large!

Solution 1

- Find the smallest decision tree consistent with any training set

Solution 2

- **Divide and conquer**
 - Choose the best attribute
 - Split the training set into subsets corresponding to that attribute
- **Recursively divide and conquer smaller training sets**

Which attribute do you split on?

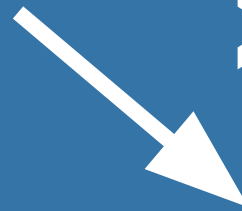
- Gives subsets with clear differences

How long is the wait?

$< 0.5h$



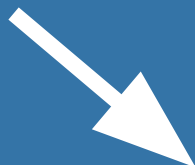
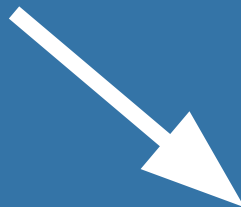
$> 0.5h$



When can we stop?

- When a subset is **100% yes** or **100% no**
- Some attributes won't be included

?



Goal

- Choose attributes that minimize tree depth
- Perfect attribute divides examples into subsets of 100% yes and 100% no

**Make Decisions
with Decision
Trees ✓**

**Next
up...**